ISO TC184/SC4/WG3 (T21) N941 **Date: 2000-05-30**

Supersedes:

Industrial automation systems and integration — Reference data library for the process industries — Part 6: Scope and methodology for developing additional reference data

ABSTRACT:

This document describes the information requirements for defining additional reference data..

KEYWORDS:

industrial data, oil and gas, process industries, facility, life-cycle, integration, overview

Project leader: ??????	Part editor: ?????
Telephone: Fax: Email:	Telephone: Fax: Email:

Comments to reader:

This document is a pre NWI working draft. Unresolved issues are indicated by boxed text.

Interim editorial guidelines, and an accompanying Word template, have been used in the preparation of this document. These guidelines apply the requirements of the ISO/IEC Directives 3, and appropriate requirements of the SC4 Supplementary Directives for ISO 10303.

Document type:	International Standard	Document subtype:	Not applicable
Document stage:	Working Draft (20)	Document language:	E
File name:	Part Tv0.9.doc	Template:	ISO SC4 latest.dot

Contents

Page

Figures

Foreword

ISO (the International Organization for Standardization) is a world wide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75% of the member bodies casting a vote.

International Standard ISO 15926-6 was prepared by Technical Committee ISO/TC184, *Industrial automation systems and integration*, Subcommittee SC4, *Industrial data*.

ISO 15926 consists of the following parts under the general title *Industrial automation systems and integration – Integration of life-cycle data for oil and gas production facilities:*

— Part 1, Overview and fundamental principles

— Part 2, Data model

— Part 4, Initial reference data;

— Part 5, Procedures for registration and maintenance of reference data;

— Part 6, Scope and methodology for developing additional reference data.

Annexes A, B and C form an integral part of this part of ISO 15926.

0 Introduction

ISO 15926 is organized as a number of parts, each published separately. This part of ISO 15926 states the information required when defining additions to the initial set of reference data given by ISO 15926-4.

0.1 Background

ISO 15926 is an International Standard for the representation of process industries facility life-cycle information. This representation is specified by a generic, conceptual data model that is suitable as the basis for implementation in a shared database or data warehouse. The data model is designed to be used in conjunction with reference data, i.e., standard instances that represent information common to a number of users, production facilities, or both. The support for a specific life-cycle activity depends on the use of appropriate reference data in conjunction with the data model.

0.2 Purpose of ISO 15926-6

The purpose of this International Standard is to define the data that shall be provided when making additions to the ISO 15926 reference data library.

0.3 Description of ISO 15926-6

ISO 15926-6 consists of:

a) a statement of scope of the standard;

b) a description of the methodology for representing new reference information using the data model of ISO 15926-2;

c) a specification of the data required to define additions to the reference data library.

0.4 Typographical conventions

The following typographical conventions are used in this International Standard.

A numbered reference enclosed in brackets (for example, “[2]”) is a reference to a document that is listed in the Bibliography.

In this International Standard the same English language words may be used to refer to an object in the real world or to a concept, and as the name of an EXPRESS data type that represents this object or concept. The following typographical convention is used to distinguish between these. If a word or phrase occurs in the same typeface as narrative text, the referent is the object or concept. If the word or phrase occurs in a bold typeface, the referent is the EXPRESS data type. Names of EXPRESS schemas also occur in a bold typeface.

The name of an EXPRESS data type may be used to refer to the data type itself, or to an instance of the data type. The distinction between these uses is normally clear from the context. If there is a likelihood of ambiguity, the phrase “entity data type” or “instance(s) of” is included in the text.

Double quotation marks “ ” denote quoted text. Single quotation marks ‘ ’ denote particular text string values.

Some components of this International Standard are available in electronic form. This access is provided through the specification of Universal Resource Locators (URLs) that identify the location of these files on the Internet.

If there is difficulty accessing these files contact the ISO Central Secretariat, or contact the ISO TC 184/SC4 Secretariat directly at: sc4sec@cme.nist.gov.

Industrial automation systems and integration — Reference data libraries for the process industries — Part 6: Scope and methodology for developing additional reference data

1 Scope

This part of ISO 15926 specifies the scope and information required when defining additions to the ISO 15926 reference data library.

The following are within the scope of this part of ISO 15926:

a) The information scope of additional reference data.

b) The methodology for representing additional information in the reference data library

c) The information requirements for defining additions to reference data.

The following are outside the scope of this part of ISO 15926:

a) Methods or guidelines for using ISO 15926-2 to represent information other than reference information

2 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this part of ISO 15926. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

ISO 10303-11:1994, *Industrial automation systems and integration — Product data representation and exchange — Part 11: EXPRESS language reference manual.*

ISO 10303-21:1994, *Industrial automation systems and integration — Product data representation and exchange — Part 21: Implementation methods: Clear text encoding of the exchange structure*

ISO 15926-2:—[1)], *Industrial automation systems and integration — Integration of life-cycle data for oil and gas production facilities — Part 2: Data model.*

ISO 15926-4:—[2)], *Industrial automation systems and integration — Integration of life-cycle data for oil and gas production facilities — Part 4: Initial reference data.*

[1] To be published, currently at CD level.

[2] To be publish, currently being developed as a NWI.

3 Terms, definitions, and abbreviations

For the purposes of this part of ISO 15926 the following terms, definitions and abbreviations apply; those taken or adapted from ISO 10303-1:1994 are repeated below for convenience.

NOTE Definitions copied verbatim from ISO 10303-1:1994 are followed by "[ISO xxxxx-y]". Definitions that have been adapted from other standards are followed by an explanatory note.

3.1
application object
a thing that can have its existence recorded

NOTE This is a different definition of this term from that found in ISO 10303-1. This definition is consistent with the use of this term in the data model defined in ISO 15926-2.

3.2
application object instance
an instance that represents a particular application object

3.3
application protocol
a part of ISO 10303 that specifies a data model that satisfies the scope and information requirements of a specific application

NOTE Adapted from ISO 10303-1.

3.4
class
A subdivision, sort or category with rules that define inclusion and or exclusion.

3.5
commodity product class
a class of manufactured item that conforms to agreed open standards

3.6
core class
a class that is a commonly used subdivision corresponding to terms used in common language

3.7
data
a representation of information in a formal manner suitable for communication, interpretation, or processing by human beings or computers

[ISO 10303-1]

3.8
defacto class
a class corresponding to common natures that are widely recognised but not formally defined or agreed

3.9
distinct facility data
facility life-cycle data that represents information that is distinct to an individual process industries facility

3.10
exchange file
a computer-interpretable format used for storing, accessing, transferring and archiving data

NOTE Adapted from the definition of "exchange structure" in ISO 10303-1.

3.11
facility life-cycle data
data that represents, in computer processable form, information about process industries production facilities

3.12
information
facts, concepts, or instructions

[ISO 10303-1]

3.13
instance
data that represents, in computer processable form, some real world thing

NOTE This is a different definition of this term from that found in ISO 10303-11.

3.14
manufactured item class
a class whose members are manufactured objects

3.15
proprietary product class
a manufactured item class where the copyright to specifications of conditions for membership is controlled in a proprietary or closed way

3.16
reference data
data, in the form of application object instances, that represents reference information.

3.17
reference data library
a managed collection of reference data

3.18
reference information
information about a ‘thing’ or type of ‘thing’ which is of common interest to many process industries facilities and users;

3.19
standards class
a class where the conditions for membership are owned/controlled by a standardisation body

4 Scope of additional reference data

Additions to the ISO 15926 reference data library shall be limited by the following:

a) The additions to the ISO 15926 reference data library shall be within the information scope of ISO 15926 and be encoded in accordance with the data model of ISO 15926-2.

b) Additions are restricted to commonly used abstractions that describe

— the physical nature of manufactured equipment and materials

— the physical nature of the environment

— functional natures of equipment and materials

— spatial features of equipment and material

— physical properties of equipment, materials

— measurement units for physical properties

— process materials

— materials of construction of facilities and facility equipment

— activities and events associated with the use of equipment and materials

— states and statuses associated with the use of equipment and materials

— information

— document annotations and symbols

c) Additions to reference data shall exclude all individual things, including

— particular facilities, equipment and materials

— geo political entities such a countries, borders, towns, and places

— particular organisations and people

— geographic and geological features such as seas, coasts, land, sedimentary basins, faults, mountains

d) Additions shall not duplicate information that is already represented in the library, but they may make additional references to the existent content.

EXAMPLE 1 If the reference data library already includes representation of the concept of 'internal combustion engine', additions to the library shall not create another representation of 'internal combustion engine', but may make references to the concept.

e) Additions shall not include information that can be reasonably derived from the reference library that includes the additions.

EXAMPLE 2 Suppose that class C is a specialization of class B and class B is a specialization of class A, then it can be derived that C is also a specialization of A. The derived information that C is a specialization of A should be omitted.

EXAMPLE 3 Relationships that are inherited from a super classes should not be explicitly defined for the subclass.

f) Additions shall be restricted to representing commonly used abstractions whose definitions are explicitly or implicitly maintained in the public domain. These include core, de facto, standards and commodity classes.

EXAMPLE 4 If the concept of measurement unit 'Knot' was not represented in the library, it could be included as a standards class by an addition to the library.

g) Abstractions that depend on proprietary definitions shall not be represented in the reference data library.

EXAMPLE 5 The specifications of a manufactured product is an abstraction whose definition is usually maintained in a closed and proprietary manner. The reference data library excludes representations of such proprietary product specifications.

5 Methodology

Additional classes can be added to the reference data register using the procedures of ISO 15926-5. Each addition shall be defined in terms of the entity types of the data model of Part 2 and encoded in accordance with the provisions of clause 6.

The use of the ISO 15926-2 data model entity types to represent some types of additional information is described below.

5.1 Physical natures of equipment and materials

Additional types of equipment and materials may be added to the reference data library.

Equipment and material types that constrain physical aspects of the membership are represented as **class_of_physical_object** that are specializations of the 'artefact' class.

The artefact classes define categories based on physical form and substance, and exclude categories based on function.

The specialization of artefact classes is defined using the **specialization_of_class** relationship. This may be defined directly with an artefact class as the super class, or more likely to an existing subclass of the artefact class. A new class must be a proper subdivision of the artefact class, meaning there must be a clear distinction allowing determination of the subset of artefacts.

EXAMPLE 1 The **class_of_physical_object** 'thrust bearing' is a specialization of the 'bearing' **class_of_physical_object**. Note the term thrust implies a particular shape and form of the artefact and hence is a proper subdivision, based on physical form.

The features, parts or other characteristics that distinguish one type of equipment or material from another are defined by relationships to the appropriate types of features, part or other characteristics.

Features and characteristics that are common to all members of a class are indicated by instances of **common_possession_of_aspect** where the possessed aspect class is either a **class_of_spatial_aspect** or a **class_of_characteristic**.

EXAMPLE 2 An orifice plate has a diameter and thickness. The range of permitted diameters and thickness for a type of plate are defined by **common_possession_of_property**. The holes of orifice plates are defined as instances of **common_spatial_aspect**.

Replaceable parts or components that are common to all members of the new class are indicated by instances of the association **common_composition_of_physical_object**, where the part refers to another **class_of_physical_object** describing the type of part.

The use of common_composition or common_possession appears to be subjective. Model issue.

EXAMPLE 3 A centrifugal pump has one or more impellers as parts. The 'centrifugal pump' class is a specialization of the 'pump' **class_of_physical_object** and is linked to the 'impeller' **class_of_physical_object** by the association **common_composition_of_physical_object**.

5.2 Equipment and material features

Additional types of equipment and material features may be added to the library.

Features of equipment or material such as holes, grooves, bends, threads, finishes, edges and ends are defined as **concept_of_spatial_aspect**. They are related to the equipment types they are features of by the association **common_possession_of_aspect.**

Features can be thought of as a space part for all the time dimension of a 4D object. Then it is apparent that Class_of_physical_feature and class_of_spatial_aspect (classes of possessed spatial aspect) overlap. Gives 2 ways of representing features. One has been chosen here.

Features may be subtyped according to their spatial degrees of freedom: **concept_of_point_in_space**; **concept_of_curve_in_space**; **concept_of_surface_in_space**; **concept_of_volume_in_space**.

Membership of feature classes may also be constrained by **class_of_property.** The resulting classes are defined as instances of the subtypes of **common_spatial_aspect**.

EXAMPLE 1 Some types of orifice plate have R24 grooves suitable to accept certain types of flange fittings. Groove is defined as **concept_of_volume_in_space**. The R24 type of groove is defined by an instance of **common_volume_in_space**, and declared as a specialization of the groove class. The dimensions of the type of groove are defined by one or more **common_property** linked to the groove type class by a **common_possession_of_property** relationship. Instances of **common_possession_of_aspect** is used to link the types of orifice plate to the types of groove.

Diameter, height, width, perimeter, apex, corner are also be features. Width, height, diameter and perimeter are linear features that are instances of **concept_of_line_in_space**.

5.3 Functional roles of equipment and materials

Additional types of functional roles may be added to the library.

Reference data distinguishes between functional and physical natures. Physical natures are defined as **class_of_physical_object** that are specializations of artefact.

Functional natures are defined by a different set of **class_of_physical_object** distinguished from physical classes by classification as members of the 'function' **class_of_class**.

Functional is a term that describes what things do. Being a member of a functional class means performing the function. The functional natures are defined independently of any physical form that might behave in the appropriate fashion.

EXAMPLE 1 An optical switch that changes the direction of a beam of light is defined as a functional **class_of_physical_object**. No reference is made to any physical form.

Links may be created to indicate which physical forms can be used for a function. The relationship …….. is used to link types of physical form to the functional types, indicating that things of that form and substance can perform this function.

Functional classes may be specialized by characteristics and properties.

Properties such as range of load, or rate of production, rate of transfer that are common to the members of a functional class are indicated by relationships **common_possession_of_property** to **common_property**.

EXAMPLE 2 A type of oil production facility may be limited to production rates less than 10,000 BPD. The type of oil production facility is represented by a functional **class_of_physical_object** that is a specialisation of the 'oil production-facility' **class_of_physical_object**. The production rate range is defined by a **common_property**, classified as a member of the 'production rate' **concept_of_property**, where the members of the **common_property** correspond to all the production rates less than or equal to 10,000 BPD.

Functional types may have other functional types as parts. This is indicated by the relationship **common_composition_of_physical_object**.

EXAMPLE 3 A type of oil production facility may be composed of risers, manifolds, water separators, gas separators, oil pumps and control systems.

5.4 Materials and substances

Types of process materials and materials of construction or substances are defined as **class_of_atomic_and_subatomic_structrure**.

EXAMPLE 1 Glass reinforced plastic is a **class_of_atomic_and_subatomic_structure**.

The name of atomic structure is misleading if this use for Wood , plywood, laminates, GRP ..... is valid. Materials of construction and process materials appear to be functional in nature, part of the life cycle of a 4D object.

The type composition of materials can be included using **common_possession_of_aspect** to refer to the component type.

How to specify relative amounts (volumetric, mass) ? Model issue.

Additional equipment and material classes that restrict or specify the type of material the members are made of can be defined by specialization of existing physical classes using a **common_possession_of_aspect** relationship to refer to the appropriate possessed **class_of_atomic_and_subatomic_structure**.

5.5 Physical properties

Additional types of physical properties of equipment and materials are defined as **concept_of_property** These refer to the physical natures of matter or energy, and exclude occupation of space and time.

EXAMPLE 1 Viscosity is a **concept_of_property**. Temperature is a **concept_of_property**.

EXAMPLE 2 Distance in space, often referred to as length is a property, whereas spatial dimensions such as length, height, width and diameter are not.

Particular magnitudes of a property are defined as **common_property** classes. The numeric scale values used to identify property magnitudes can be given as **common_property_value** classes.

EXAMPLE 3 The particular degree of temperature known as 30 deg C is a **common_property**, classified as the temperature **concept_of_property**. The range of temperatures 20 to 30 deg C is a **common_property** class.

Concept of property should be remodelled as subtype of class_of_class. Model issue.

Physical properties such as operating pressure, inlet temperature may be defined as specializations of the appropriate **concept_of_property**, adding the appropriate qualifier.

Inlet temperature should refer to the 'inlet' class. It is not clear what relationship should be used for this. Temperature should be a class of class. But inlet is a functional class of physical object. There can be a 50deg C inlet class. How is it related to say a compressor – compressor has a part that is a member (part member) ?

5.6 Dimensions and Units of Measure

Additional dimensions and units of measure may be added as **concept_of_property_value** classes. Each new unit of measure should be a specialisation of an appropriate dimensional quantity. There is no need to define any conversions to other units of the same dimensional type as these can be derived if the dimensional type is given in terms of the base dimensions.

EXAMPLE 1 The unit of speed known as the Knot can be added as a new instance of **concept_of_property_value** that is a specialisation of the speed dimension (LT^{-1}).

New dimensional quantities should be defined in terms of the base dimensions, allowing units of measures to be derived. The relationship **common_factorization_of_property_value** enables the dimensional factors of a new dimensional quantity to be defined.

EXAMPLE 2 The speed dimensional quantity (LT^{-1}) is defined in terms of the base dimensions L and T by two instances of **common_factorization_of_property_value**, one to T with exponent of -1 and the other to L with an exponent of +1.

The dimensions and units of measure ought not to be related by specialisation but by classification of class. Model issue.

5.7 Spatial characteristics

Spatial characteristics of equipment and materials such as heights, widths, perimeters, axes, boundaries and sections may be defined to be used in the specialization of artefacts. They are similar in nature to features, and are considered to be spatial aspects of 4D objects and handled accordingly.

Types of spatial characteristic may be defined as one of the subtypes of **concept_of_spatial_aspect**: **concept_of_point_in_space**, **concept_of_curve_in_space**, **concept_of_surface_in_space** and **concept_of_volume_in_space**.

EXAMPLE 1 The concept of 'diameter' is represented as a **concept_of_curve_in_space**. The concept of 'impeller diameter' is represented as a **concept_of_curve_in_space** that is a specialization of the 'diameter' class and linked by **common_possession_of_aspect** to the possessor class 'impeller'.

Types of spatial characteristics may be constrained in size by referencing the relevant property classes. Types of property that are relevant to linear spatial characteristics are length and direction.

There is no obvious place for shape classes e.g. for lines: straight, circle parabolic, open, closed. Shape is a class of class. Straight is a type of 1D space and a member of the shape class_of_class. Spaces for all time is needed.

EXAMPLE 2 Impeller diameters of 10 mm are represented as a **common_curve_in_space** that is a specialization of the impeller diameter class and linked by **common_possession_of_aspect** to a common_property class of 10mm.

5.8 Events and activities

Events, event effects and activities need to be sorted. Activity is said to be a fusion of events, and therefore still an event (a 4D object delta t = 0. Cause and effect become whole part or possession associations. Modelling for involvement and role in life is done by simple events being part of 4D object and part of the activity.

Types of events that effect plant, equipment and materials are represented by the subtype **concept_of_point_in_time**. These can include both simple events that effect one particular object and complex series of events that affect many objects.

EXAMPLE 1 'Accident', 'failure', 'shut down', and 'start up' are types of event represented as instances of **concept_of_point_in_time**.

Activities are complex events, made up from simple events.

EXAMPLE 2 Distillation, fabrication, control, measure, calibration are all [.....].

The make up of types of complex events, if known, can be indicated by [.....], where the part refers to the simple event.

EXAMPLE 3 'Switch on' is an event may be part of a more complex event or activity called 'start up'.

5.9 States and status

If state is thought of as finite time slice or temporal aspect of the entire spatial extent of a 4D physical object, State classes and classes of possessed temporal aspects overlap. Two ways for the same thing. Needs resolving. Here we choose one.

Types of state, such as condition, health, functional, physical, and operational are represented as **concept_of_period_of_time**.

The distinctions or values of a type of state are represented as **common_period_of_time** and are made members of the **concept_of_period_of_time** using instances of the **classification_of_class** relationship.

Concept as used here, would be a class_of_class, not class_of_aspect.

Why are is phase modelled separately from state. Phase is a class whose members include the classes liquid, gas, solid, crystalline, amorphous....

EXAMPLE 1 The operational state of a motor may be ready or failed. 'Operational state' is represented as a **concept_of_period_of_time** class and the conditions of 'ready' and 'failed' are represented as **common_period_of_time** classes that are classified as members of the 'operational state'

5.10 Time and periods of time

Time and periods of time that are not aspects are absent from V3.1 New entity types Instant_of_time and Period_of_time for representing times and classes of time such as months, days, quarters are should be added. Types of periods of time such as years, months, days, quarters and half years are represented as [.....].

5.11 Symbols and text annotation

Descriptions of the form of types of symbols and annotation (writing characters) may be added as **class_of_encoded_information**.

Common_encoded_information classes are used to identify particular physical forms of symbols and text.

Common text forms are defined as **common_textual_encoded_information**, where an example of the text is held in the representation attribute.

The representation is better attached to the symbol function as it is just one of many representations.

Usage of a particular physical form of a symbol or annotation as a symbol or reference is indicated by creating a **common_symbol** class and by referencing the **common_encoded_information** using **common_expression_of_symbol_by_encoded_information**.

The **common_symbol** class may be classified as a name, symbol or reference, and linked to the application object it refers to by a instance of **common_reference_to_application_object**.

Categories of the forms of symbols and or characters are defined by classifying each symbol form or character as a member of an appropriate **concept_of_encoded_information.**

5.12 Documents and information

The media presentation of documents are physical in nature and may give rise to classes of physical objects that qualify either the physical form of the document or its use or purpose.

The information they convey may be defined as **common_information** classes. Types of information are defined as **concept_of_information** classes, used to classify **common_information** classes.

EXAMPLE 1 ANSI standard information is a **concept_of_information** whose members are the information conveyed in the various ANSI standard documents. Note the information common to all renderings or copies of an ANSI document is a class, each individual interpretation of a particular document copy being a class member.

A particular form of rendering or presentation of the information may be defined as an instance of **common_encoded_information** linked by the relationship **common_expression_of_information_by_encoded_information**.

6 Data requirements

The data requirements for defining additions to the ISO 15926 Reference Library are formally described by the EXPRESS language information model given in Annex C.

For the purposes of submitting additions to the ISO 15926-4 reference data library, the additions shall be defined as an encoding of the EXPRESS data types defined in Annex C using the clear text encoding method specified by ISO 10303-21.

The following additional qualifications shall be applied to the information required in the Part 21 header. The entity type names refer to the Header Section Schema of ISO 10303-21.

— **file_description.description** shall contain a textual description of the nature of the additions to register defined in the following file;

— **file_description.implementation_level** shall contain character ‘1’ as the first character of the string;

— **file_name.name** shall contain the name, defined by the submitter, for the set of proposed register additions;

— **file_name.time_stamp** shall contain the date and time at which the file contents was created. The date and time shall use the UTC time system;

— **file_name.author** shall contain the name and address of the person responsible for the submission;

— **fle_name.organization** shall contain the organisation with whom the author is associated;

— **file_name.preprocessor_version, originating_system, authorization** shall be blank.

6.1 Class definition

Additional classes or abstractions to be added to the reference data library shall be defined as instances of **class_definition**. A **class_definition** shall refer to one or more **class_type** where the **class_type** is an enumerated type of some the ISO 15926-2 entity types (see ISO 15926-2 for definitions of the types).

The information required for each additional class is defined by the attributes of **class_definition**:

a) The type or types of the class.

b) A name of the class. The following rules apply:

— names shall use English words;

— names shall be singular;

— names shall not contain abbreviations, underscores or other punctuation marks;

— the name text characters shall be upper or lower case;

— names of characters shall not include italic, bolded or underlined characters;

— words of a name shall be ordered according to the rules of the English language.

c) A textual definition of the class. The following rules apply:

— definitions shall be in English;

— class definitions shall refer to classes that are direct generalizations and describe the new constraints that give the new subdivision.

EXAMPLE The definition of the class green widget might read “A green widget is a widget of the colour green.

d) Values for any additional attributes of the class, the attributes are dependent on the type of the class

The meaning and data requirements of d) above depend on the type of attribute and are defined by ISO 15926-2.

6.2 Class relationship definition

New relationships between classes may be defined as instances of **class_relationship_definition**. The related classes may be new or already defined within the register. Data representing valid relationships of a class should be defined where ever the relationships are known.

A **class_relationship_definition** shall refer to one or more **class_relationship_type** where the **class_relationship_type** is an enumerated type of some the ISO 15926-2 entity types (see ISO 15926-2 for definitions of the types).

The information required for each additional relationship is defined by the attributes of **class_relationship_definition**:

a) The type of relationship.

b) The references for each role of the relationship. These may be new or previously registered classes. The classes that may be referenced by a relationship definition are constrained by the type of relationship. The types of class that can be referenced by a type of relationship is given by ISO 15926-2.

c) Values for any additional attributes of the relationship, the required attributes are dependent on the type of relationship.

The meaning and data requirements of c) above depend on the type of relationship attribute and are defined by ISO 15926-2.

6.3 References to registered classes

References to classes that are already represented in the reference data library are defined as instances of **class_reference**. A **class_reference** is identified by the register identifier for the class. The register identifier of a class is defined by ISO 15926-4.

Annex A
(normative)

Information object registration method

In order to provide unambiguous identification of schemas and other information objects in an open information system, this International Standard employs the registration technique defined in ISO/IEC 8824-1.

NOTE This registration technique is equivalent to that defined in 4.3 of ISO 10303-1 for information objects standardized in ISO 10303.

This technique identifies objects by their assignment to a tree structure whose root is ISO itself. Each node in the tree is identified by a sequence of integers corresponding to the index of the leaf under each node. Nodes that identify agencies that can further specify inferior nodes are called registration authorities. There is provision in this technique for having registration provided by national bodies and other identified organizations (including private corporations). A registration authority is automatically granted to the technical committee or subcommittee that prepares a standard in order to identify objects within the standard.

Thus, ISO 15926 is identified by the object identifier:

```
{ 1 0 15926 }
```

Here the initial 1 indicates ISO; the 0 following it identifies the object as a standard, and the number following that is the number of the standard. ISO/IEC 8824-1 also defines identifiers to stand in the place of these numbers; thus 'iso' has the value 1 and 'standard' has the value 0. For multi-part standards, the next number is required to be the part number. Thus, this part of ISO 15926 is identified by the object identifier:

```
{ iso standard 15926 part(r) }
```

Here, the value of the part number is given explicitly, but the notation allows us to associate a term with this value, thereby providing some semantics. The notation for values of this type is defined in clause 28 of ISO/IEC 8824-1, and the predefined assignments are specified in annex B of ISO/IEC 8824-1.

For the purposes of identifying information objects unambiguously within an open information system, ISO 15926 adopts the following conventions:

a) The value following the part number shall be version number. By convention, the value of the version number of the first edition shall be 1. The value 0, if used at all, is reserved to refer to DIS documents.

b) The value following the version number is used to identify the type of information object defined within the part. The value 1 shall indicate that the object so identified is a schema.

c) The value following the object type is an integer that identifies the instance of the object type so identified.

d) To meet the syntactic requirements of ISO/IEC 8824-1, replace each occurrence of underscore character "_" in a schema name with a hyphen when defining this value.

EXAMPLE The oil_and_gas_production_facilities schema defined in ISO 15926-2 can be identified by the value

```
{ iso standard 15926 part(2) version(1) object(1) oil-and-gas-production-facilities
                                    (1) }
```

Annex B
(normative)

Information object registration

To provide for unambiguous identification of an information object in an open system, the object identifier

```
{iso standard 15926 part{r} version {1}}
```

is assigned to this part of ISO 15926. The meaning of this value is defined in ISO/IEC 8824-1, and is described in Annex A.

This is the object identifier that will apply to the published (IS) version of this part of ISO 15926.

Annex C
(normative)

Information model for reference data additions

The data requirements are defined by the following information model, defined in EXPRESS

```
(*
```

EXPRESS specification:

```
*)
SCHEMA REGISTER;
(*
```

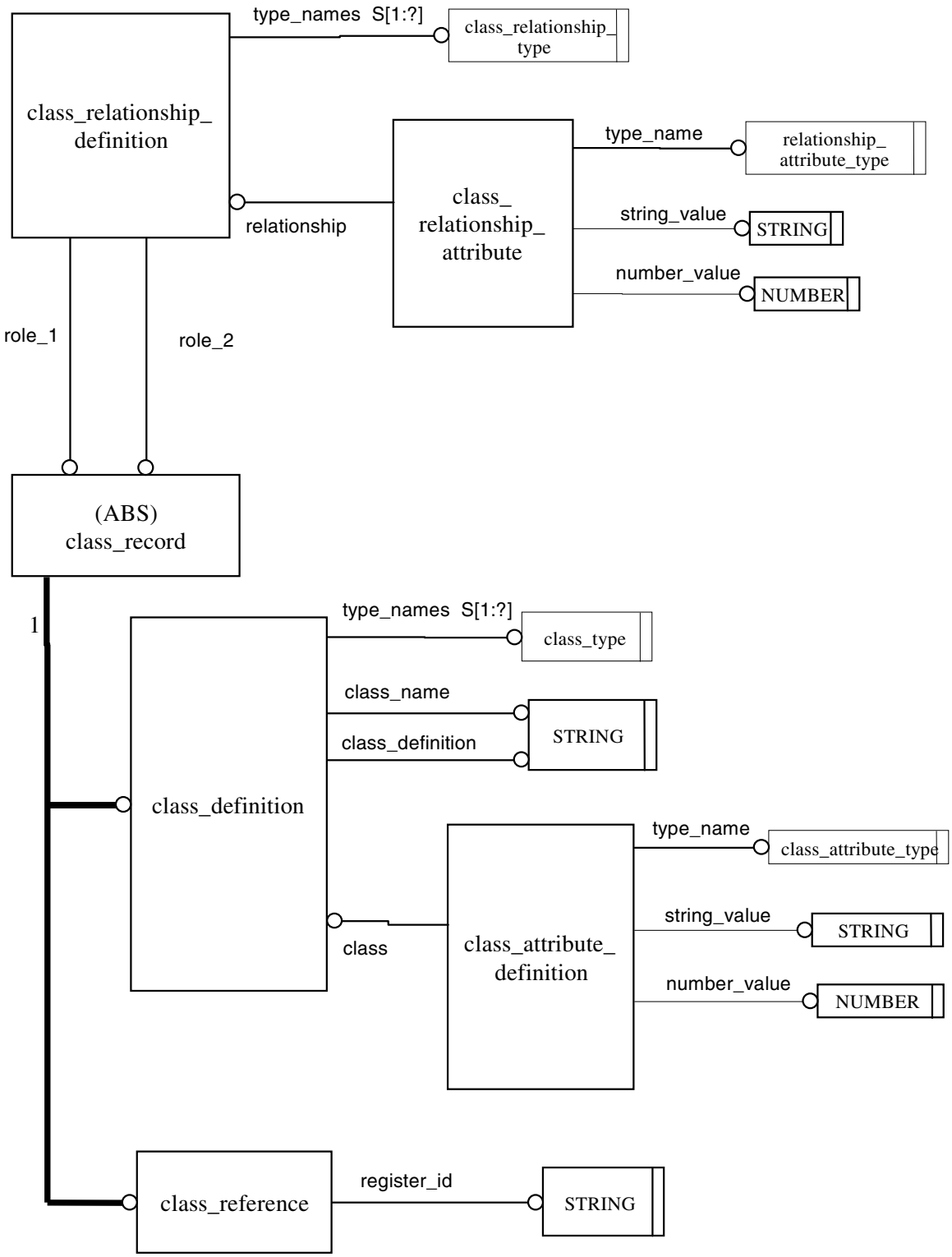

Figure 1 - REGISTER schema EXPRESS-G diagram 1 of 1

C.1 Enumerated types

C.1.1 class_attribute_type

EXPRESS specification:

```
*)
   TYPE class_attribute_type = ENUMERATION OF
      (NOMINAL_VALUE,
      LOWER_TOLERANCE,
      UPPER_TOLERANCE,
      LOWER_BOUND,
      UPPER_BOUND,
      YEAR,
      MONTH,
      DAY,
      HOUR,
      MINUTE,
      SECOND);
   END_TYPE;
(*
```

C.1.2 class_relationship_type

EXPRESS specification:

```
*)
   TYPE class_relationship_type = ENUMERATION OF
      (SPECIALIZATION_OF_CLASS,
      COMMON_POSSESSION_OF_ASPECT,
      COMMON_INFORMATION_ABOUT_MEMBER_OF_CLASS,
      COMMON_REFERENCE_TO_MEMBER_OF_CLASS,
      COMMON_COMPOSITION_OF_ACTIVITY,
      COMMON_TEMPORAL_SEQUENCE_OF_ACTIVITY,
      COMMON_INVOLVEMENT_OF_PHYSICAL_OBJECT_IN_ACTIVITY,
      COMMON_POSSESSION_OF_PROPERTY,
      COMMON_POSSESSION_OF_VALUE_BY_PROPERTY,
      COMMON_POSSESSION_OF_END_POINT_BY_PERIOD_OF_TIME,
      COMMON_POSSESSION_OF_START_POINT_BY_PERIOD_OF_TIME,
      COMMON_POSSESSION_OF_VALUE_BY_POINT_IN_TIME,
      COMMON_POSSESSION_OF_SPATIAL_ASPECT,
      COMMON_POSSESSION_OF_QUALITY,
      COMMON_POSSESSION_OF_STATE,
      COMMON_POSSESSION_OF_TEMPORAL_ASPECT_BY_STATE,
      COMMON_INCLUSION_OF_INFORMATION,
      COMMON_POSSESSION_OF_INFORMATION_BY_ASPECT,
      COMMON_CONVERSION_OF_ENCODED_INFORMATION,
      COMMON_EXPRESSION_OF_INFORMATION_BY_ENCODED_INFORMATION,
      COMMON_EXPRESSION_OF_SYMBOL_BY_ENCODED_INFORMATION,
      COMMON_FACTORIZATION_OF_PROPERTY_VALUE,
      COMMON_LINEAR_CONVERSION_OF_PROPERTY_VALUE,
      COMMON_REFERENCE_TO_APPLICATION_OBJECT,
      COMMON_COMPOSITION_OF_PHYSICAL_OBJECT,
      COMMON_CONNECTION_OF_PHYSICAL_OBJECT,
      COMMON_INSTALLATION_OF_PHYSICAL_OBJECT_FOR_FUNCTIONAL_PHYSICAL_OBJECT,
      COMMON_ROLE_IN_LIFE_OF_PHYSICAL_OBJECT,
      CLASSIFICATION_OF_CLASS,
      COMMON_PHYSICAL_FEATURE_PART_OF_PHYSICAL_OBJECT,
      COMMON_ASSERTING_ASSOCIATION,
      COMMON_DENYING_ASSOCIATION);
   END_TYPE;
(*
```

C.1.3 class_type

EXPRESS specification:

```
*)
   TYPE class_type = ENUMERATION OF
      (CLASS_OF_ACTIVITY,
      CLASS_OF_CLASS,
      COMMON_PROPERTY,
      CONCEPT_OF_PROPERTY,
      COMMON_POINT_IN_TIME,
      CONCEPT_OF_POINT_IN_TIME,
      COMMON_PERIOD_IN_TIME,
      CONCEPT_OF_PERIOD_IN_TIME,
      COMMON_POINT_IN_SPACE,
      CONCEPT_OF_POINT_IN_SPACE,
      COMMON_CURVE_IN_SPACE,
      CONCEPT_OF_CURVE_IN_SPACE,
      COMMON_SURFACE_IN_SPACE,
      CONCEPT_OF_SURFACE_IN_SPACE,
      COMMON_VOLUME_IN_SPACE,
      CONCEPT_OF_VOLUME_IN_SPACE,
      CLASS_OF_ATOMIC_AND_SUBATOMIC_STRUCTURE,
      CLASS_OF_PHASE,
      COMMON_QUALITY,
      CONCEPT_OF_QUALITY,
      COMMON_STATE,
      COMMON_INFORMATION,
      CONCEPT_OF_INFORMATION,
      COMMON_ENCODED_INFORMATION,
      CONCEPT_OF_ENCODED_INFORMATION,
      COMMON_NOMINAL_PROPERTY_VALUE,
      COMMON_BOUNDED_PROPERTY_VALUE,
      COMMON_LOWER_BOUNDED_PROPERTY_VALUE,
      COMMON_UPPER_BOUNDED_PROPERTY_VALUE,
      CONCEPT_OF_PROPERTY_VALUE,
      COMMON_CALENDAR_TIME_VALUE,
      CONCEPT_OF_POINT_IN_TIME_VALUE,
      COMMON_SYMBOL,
      CONCEPT_OF_SYMBOL,
      CLASS_OF_ORGANIZATION,
      CLASS_OF_INANIMATE_PHYSICAL_OBJECT,
      CLASS_OF_PHYSICAL_OBJECT,
      CONCEPT_OF_STATE,
      COMMON_TEXTUAL_ENCODED_INFORMATION,
      CLASS_OF_PHYSICAL_FEATURE);
   END_TYPE;
(*
```

C.1.4 relationship_attribute_type

EXPRESS specification:

```
*)
   TYPE relationship_attribute_type = ENUMERATION OF
      (EXPONENT,
      MULTIPLIER,
      OFFSET,
      ROLE_1_LIFE_LOWER_CARDINALITY,
      ROLE_1_LIFE_UPPER_CARDINALITY,
      ROLE_2_LIFE_LOWER_CARDINALITY,
      ROLE_2_LIFE_UPPER_CARDINALITY,
      ROLE_1_SIMULTANEOUS_LOWER_CARDINALITY,
      ROLE_1_SIMULTANEOUS_UPPER_CARDINALITY,
      ROLE_2_SIMULTANEOUS_LOWER_CARDINALITY,
      ROLE_2_SIMULTANEOUS_UPPER_CARDINALITY);
   END_TYPE;
(*
```

C.2 Entity types

C.2.1 class_attribute_definition

A record defining an attribute of the new class entity type to be added to the register. The attributes that can be added are constrained by the **class_definition.class_type**. The valid types of attribute for each **class_type** are defined by ISO 15926-2 data model.

EXPRESS specification:

```
*)
   ENTITY class_attribute_definition;
      class                          :class_definition;
      number_value                   :OPTIONAL NUMBER;
      string_value                   :OPTIONAL STRING;
      type_name                      :class_attribute_type;
   END_ENTITY;
(*
```

Attribute definitions:

class : The **class_definition** that the **class_attribute_definition** is part.

number_value : A number, either real or integer, defining the attribute value. The number_value should be used only when the type name **class_attribute** is declared as a NUMBER, REAL or INTEGER type in ISO 15926-2.

string_value : A text string defining the attribute value. The **string_value** should be used only when the type name **class_attribute** is declared as a STRING type in ISO 15926-2.

type_name : The attribute name that the **class_attribiute_definition** record provides a value for. The valid attribute names are constrained by the **class_type** of the **class_definition**.

C.2.2 class_definition

A record defining a new class to be added to the register.

EXPRESS specification:

```
*)
   ENTITY class_definition
      SUBTYPE OF(class_record);
      class_definition               :STRING;
      class_name                     :STRING;
      type_names                     :SET [1:?] OF class_type;
   END_ENTITY;
(*
```

Attribute definitions:

class_definition : A textual definition of the new class.

class_name : The name of the new class.

type_names : The names of the ISO 156926-2 entity types the class defined by the class definition is an instance of.

C.2.3 class_record

A record about a class. A class record must be either a **class_reference** for a class already within the register, or

a **class_definition** for a class to be added to the register.

EXPRESS specification:

```
*)
   ENTITY class_record
      ABSTRACT SUPERTYPE OF (ONEOF(class_definition, class_reference));
   END_ENTITY;
(*
```

C.2.4 class_reference

A record defining a reference to a class already maintained in the register.

EXPRESS specification:

```
*)
   ENTITY class_reference
      SUBTYPE OF(class_record);
      register_id                    : STRING;
   END_ENTITY;
(*
```

Attribute definitions:

register_id : The register identifier of a class that is being referred to in the extension data.

C.2.5 class_relationship_attribute

A record defining an attribute value of the new class relationship to be added to the register. The attributes that can be added are constrained by the **class_relationship_definition.type_name**. The valid types of attribute for each **class_relationship_type** are defined by ISO 15926-2 data model.

EXPRESS specification:

```
*)
   ENTITY class_relationship_attribute;
      number_value                   : OPTIONAL NUMBER;
      relationship                   : class_relationship_definition;
      string_value                   : OPTIONAL STRING;
      type_name                      : relationship_attribute_type;
   END_ENTITY;
(*
```

Attribute definitions:

number_value : A number, either real or integer, defining the attribute value. The number_value should be used only when the data type of the **relationship_attribute_type** is declared as a NUMBER, REAL or INTEGER type in ISO 15926-2.

string_value : A text string defining the attribute value. The **string_value** should be used only when the type name **class_realtionship_attribute** is declared as a STRING type in ISO 15926-2.

type_name : The attribute name that the **class_relationship_attribute** record provides a value for. The valid attribute names are constrained by the **class_relationship_type** of the **class_relationship_definition**.

C.2.6 class_relationship_definition

A record defining a relationship between two classes. The classes may be new, or already part of the register data.

EXPRESS specification:

```
*)
   ENTITY class_relationship_definition;
      role_1                          : class_record;
      role_2                          : class_record;
      type_names                      : SET [1:?] OF class_relationship_type;
   END_ENTITY;
(*
```

Attribute definitions:

role_1 : The first class involved in the **class_relationship_definition**. The meaning of Role_1 depends on the type of class relationship and are defined in ISO 15926-2.

The valid entity types of the **class_record** referred to by role_1 is constrained by the **class_relationship_type** of the **class_record**.

role_2 : The first class involved in the **class_relationship_definition**. The meaning of Role_1 depends on the type of class relationship and are defined in ISO 15926-2.

The valid entity types of the **class_record** referred to by role_2 is constrained by the **class_relationship_type** of the **class_record**.

type_names : The names of the ISO 156926-2 entity types the relationship defined by the **class_relationship_definition** is an instance of.

```
*)
END_SCHEMA;
```

Bibliography

Index